

**Ανοικτά Ακαδημαϊκά Μαθήματα**

**Τεχνολογικό Εκπαιδευτικό Ίδρυμα Αθήνας**

**Βάσεις Δεδομένων ΙΙ (Ε)**

**Ενότητα 14:** Επισκόπηση της Διαχείρισης triggers στο περιβάλλον mySQL και σε Oracle

Χ. Σκουρλάς, Α. Τσολακίδης, Δ. Βάσσης

Τμήμα Μηχανικών Πληροφορικής ΤΕ

|  |  |
| --- | --- |
| Το περιεχόμενο του μαθήματος διατίθεται με άδεια Creative Commons εκτός και αν αναφέρεται διαφορετικά | Το έργο υλοποιείται στο πλαίσιο του Επιχειρησιακού Προγράμματος «Εκπαίδευση και Δια Βίου Μάθηση» και συγχρηματοδοτείται από την Ευρωπαϊκή Ένωση (Ευρωπαϊκό Κοινωνικό Ταμείο) και από εθνικούς πόρους. |

Περιεχόμενα

[1. Επισκόπηση της Διαχείρισης triggers στο περιβάλλον mySQL και σε Oracle 3](#_Toc401565476)

[1.1 Υποδείξεις για τη διαχείριση triggers στο περιβάλλον mySQL 3](#_Toc401565477)

[1.2 Δημιουργία βάσης δεδομένων με χρήση mySQL 3](#_Toc401565478)

[1.3 Διαχείριση triggers στο περιβάλλον Oracle PL/SQL 12](#_Toc401565479)

[1.4 Κάποιες άλλες χρήσιμες υποδείξεις για τη διαχείριση των triggers στο προϊόν της Oracle 21](#_Toc401565480)

Στόχος του εργαστηρίου είναι η επισκόπηση της χρήσης triggers για τη δημιουργία εφαρμογών στο περιβάλλον mySQL αλλά και τη διαχείριση βάσεων δεδομένων. Μετά την επεξεργασία του εργαστηρίου ο ενδιαφερόμενος θα έχει κατανοήσει τα θέματα της χρήσης triggers και δημιουργίας εφαρμογής βάσεων δεδομένων. Τέλος, ο ενδιαφερόμενος θα έχει κατανοήσει τα θέματα της χρήσης triggers και δημιουργίας εφαρμογής βάσεων δεδομένων και στο περιβάλλον της Oracle.

 **Λέξεις κλειδιά:** triggers, mySQL, Oracle

# Επισκόπηση της Διαχείρισης triggers στο περιβάλλον mySQL και σε Oracle

## Υποδείξεις για τη διαχείριση triggers στο περιβάλλον mySQL

Έστω το απλοποιημένο σύστημα διαχείρισης στοιχείων προσωπικού στο περιβάλλον της mySQL που περιλαμβάνει τους πίνακες: EMP, DEPT, USER.

## Δημιουργία βάσης δεδομένων με χρήση mySQL

DROP DATABASE personnel;

CREATE DATABASE personnel;

USE personnel;

CREATE TABLE Dept(DEPTNO INT(2) NOT NULL,

 DNAME VARCHAR(14), LOC VARCHAR(14),

 NO\_OF\_EMPLOYEES INT(3),

 PRIMARY KEY(DEPTNO));

CREATE TABLE Emp(EMPNO INT(4) NOT NULL AUTO\_INCREMENT,

 ENAME VARCHAR(10), JOB VARCHAR(9),

 SAL FLOAT(7,2),

 DEPTNO INT(2), PRIMARY KEY(EMPNO),

 FOREIGN KEY(DEPTNO) REFERENCES Dept(DEPTNO));

CREATE TABLE user(

 uname text,

 upass text,

 Uid int(11),

 Uphone varchar(45), Ucity varchar(45));

INSERT INTO `user` VALUES ('admin','1234',1,NULL,NULL);

 Select \* from user;

Οι πίνακες emp, dept παρατίθενται με ενδεικτικό δείγμα δεδομένων.

INSERT INTO Dept(DEPTNO, DNAME, LOC)

 VALUES (10, 'ACCOUNTING', 'NEW YORK');

INSERT INTO Dept(DEPTNO, DNAME, LOC)

 VALUES (20, 'RESEARCH', 'DALLAS');

INSERT INTO Dept(DEPTNO, DNAME, LOC)

 VALUES (30, 'SALES', 'CHICAGO');

INSERT INTO Dept(DEPTNO, DNAME, LOC)

 VALUES (40, 'OPERATIONS', 'BOSTON');

Select \* from dept;

|  |  |  |  |
| --- | --- | --- | --- |
| **DEPTNO** | **DNAME** | **LOC** | NO\_OF\_EMPLOYEES |
| 10 | ACCOUNTING | NEW YORK |  |
| 20 | RESEARCH | DALLAS |  |
| 30 | SALES | CHICAGO |  |
| 40 | OPERATIONS | BOSTON |  |

 4 rows selected. 0,06 seconds

INSERT INTO Emp( ENAME, JOB, SAL, DEPTNO) VALUES ( 'SMITH', 'CLERK', 800, 20);

INSERT INTO Emp( ENAME, JOB, SAL, DEPTNO) VALUES ( 'ALLEN', 'SALESMAN', 1600, 30);

INSERT INTO Emp( ENAME, JOB, SAL, DEPTNO) VALUES ( 'WARD', 'SALESMAN', 1250, 30);

INSERT INTO Emp( ENAME, JOB, SAL, DEPTNO) VALUES ( 'JONES', 'MANAGER', 2975, 20);

INSERT INTO Emp( ENAME, JOB, SAL, DEPTNO) VALUES ( 'MARTIN', 'SALESMAN', 1250, 30);

INSERT INTO Emp( ENAME, JOB, SAL, DEPTNO) VALUES ( 'BLAKE', 'MANAGER', 2850, 30);

INSERT INTO Emp( ENAME, JOB, SAL, DEPTNO) VALUES ( 'CLARK', 'MANAGER', 2450, 10);

INSERT INTO Emp( ENAME, JOB, SAL, DEPTNO) VALUES ( 'SCOTT', 'ANALYST', 3000, 20);

INSERT INTO Emp( ENAME, JOB, SAL, DEPTNO) VALUES ( 'KING', 'PRESIDENT', 5000, 10);

INSERT INTO Emp( ENAME, JOB, SAL, DEPTNO) VALUES ( 'TURNER', 'SALESMAN', 1500, 30);

INSERT INTO Emp( ENAME, JOB, SAL, DEPTNO) VALUES ( 'ADAMS', 'CLERK', 1100, 20);

INSERT INTO Emp( ENAME, JOB, SAL, DEPTNO) VALUES ( 'JAMES', 'CLERK', 950, 30);

INSERT INTO Emp( ENAME, JOB, SAL, DEPTNO) VALUES ( 'FORD', 'ANALYST', 3000, 20);

INSERT INTO Emp( ENAME, JOB, SAL, DEPTNO) VALUES ( 'MILLER', 'CLERK', 1300, 10);

INSERT INTO Emp( ENAME, JOB, SAL, DEPTNO) VALUES ( 'BATES', 'ANALYST', 1300, NULL);

Select \* from emp;

|  |  |  |  |
| --- | --- | --- | --- |
|  ENAME | JOB | SAL | DEPTNO |
| SMITH | CLERK | 800 | 20 |
| ALLEN | SALESMAN | 1600 | 30 |
| WARD | SALESMAN | 1250 | 30 |
| JONES | MANAGER | 2975 | 20 |
| MARTIN | SALESMAN | 1250 | 30 |
| BLAKE | MANAGER | 2850 | 30 |
| CLARK | MANAGER | 2450 | 10 |
| SCOTT | ANALYST | 3000 | 20 |
| KING | PRESIDENT | 5000 | 10 |
| TURNER | SALESMAN | 1500 | 30 |
| ADAMS | CLERK | 1100 | 20 |
| JAMES | CLERK | 950 | 30 |
| FORD | ANALYST | 3000 | 20 |
| MILLER | CLERK | 1300 | 10 |
| BATES | ANALYST | 1300 | -  |

15 rows selected. 0,12 seconds

DROP TRIGGER emp\_insert;

DROP TRIGGER emp\_delete;

DROP TRIGGER emp\_update;

DELIMITER //

CREATE TRIGGER emp\_insert

AFTER INSERT ON emp

FOR EACH ROW

BEGIN

 UPDATE dept

 SET no\_of\_employees = IFNULL(no\_of\_employees, 0) + 1

 WHERE deptno = NEW.deptno;

END //

DELIMITER ;

DELIMITER //

CREATE TRIGGER emp\_delete

AFTER DELETE ON emp

FOR EACH ROW

BEGIN

 UPDATE dept

 SET no\_of\_employees = IFNULL(no\_of\_employees, 0) - 1

 WHERE deptno = OLD.deptno;

END //

DELIMITER ;

DELIMITER //

CREATE TRIGGER emp\_update

AFTER UPDATE ON emp

FOR EACH ROW

BEGIN

 UPDATE dept

 SET no\_of\_employees = NVL(no\_of\_employees,0) + 1

 WHERE deptno = NEW.deptno;

 UPDATE dept

 SET no\_of\_employees = IFNULL(no\_of\_employees, 0) - 1

 WHERE deptno = OLD.deptno;

END //

DELIMITER ;

**Ακολουθούν δοκιμές / έλεγχοι.**

INSERT INTO dept VALUES(11,'Belle Epoque', 'PARIS', 0);

INSERT INTO dept VALUES(12,'Rasors edge', 'PARIS', 0);

INSERT INTO emp(empno,ename,deptno) VALUES (102,'Luers',11);

INSERT INTO emp(empno,ename,deptno) VALUES (103,'Atwood',11);

INSERT INTO emp(empno,ename,deptno) VALUES (104,'Gennick',12);

SELECT \* FROM dept WHERE deptno IN (11,12);

DELETE FROM emp WHERE empno = 103;

SELECT \* FROM dept WHERE deptno IN (11,12);

UPDATE emp

SET deptno = 11

WHERE empno = 104;

SELECT \* FROM dept WHERE deptno IN (11,12);

SELECT \* FROM dept;

SELECT \* FROM emp;

Διόρθωση τιμών της στήλης NO\_OF\_EMPLOYEES.

UPDATE dept

SET no\_of\_employees = (SELECT COUNT(\*)

 FROM emp

 WHERE emp.deptno = dept.deptno);

## Διαχείριση triggers στο περιβάλλον Oracle PL/SQL

Έστω το απλοποιημένο σύστημα διαχείρισης στοιχείων προσωπικού που συνοδεύει το προϊόν της Oracle. Θα δοθούν κάποια βοηθητικά παραδείγματα που δουλεύουν με παρόμοια στοιχεία με αυτά των γνωστών πινάκων EMP, DEPT της Oracle.

**Σημείωση**

Οι πίνακες emp, dept παρατίθενται με ενδεικτικό δείγμα δεδομένων.

SELECT \* FROM EMP

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| EMPNO | ENAME | JOB | MGR | HIREDATE | SAL | COMM | DEPTNO |
| 7369 | SMITH | CLERK | 7902 | 17/12/80 | 800 | -  | 20 |
| 7499 | ALLEN | SALESMAN | 7698 | 20/02/81 | 1600 | 300 | 30 |
| 7521 | WARD | SALESMAN | 7698 | 22/02/81 | 1250 | 500 | 30 |
| 7566 | JONES | MANAGER | 7839 | 02/04/81 | 2975 | -  | 20 |
| 7654 | MARTIN | SALESMAN | 7698 | 28/10/81 | 1250 | 1400 | 30 |
| 7698 | BLAKE | MANAGER | 7839 | 01/05/81 | 2850 | -  | 30 |
| 7782 | CLARK | MANAGER | 7839 | 09/06/81 | 2450 | -  | 10 |
| 7788 | SCOTT | ANALYST | 7566 | 19/04/87 | 3000 | -  | 20 |
| 7839 | KING | PRESIDENT | -  | 17/11/81 | 5000 | -  | 10 |
| 7844 | TURNER | SALESMAN | 7698 | 08/10/81 | 1500 | 0 | 30 |
| 7876 | ADAMS | CLERK | 7788 | 23/05/87 | 1100 | -  | 20 |
| 7900 | JAMES | CLERK | 7698 | 03/12/81 | 950 | -  | 30 |
| 7902 | FORD | ANALYST | 7566 | 03/12/81 | 3000 | -  | 20 |
| 7934 | MILLER | CLERK | 7782 | 23/01/82 | 1300 | -  | 10 |
| 7999 | BATES | ANALYST | 7566 | 23/01/04 | 1300 | -  | -  |

15 rows selected. 0,12 seconds

SELECT \* FROM DEPT

|  |  |  |
| --- | --- | --- |
| DEPTNO | DNAME | LOC |
| 10 | ACCOUNTING | NEW YORK |
| 20 | RESEARCH | DALLAS |
| 30 | SALES | CHICAGO |
| 40 | OPERATIONS | BOSTON |

4 rows selected. 0,06 seconds

Αν θέλετε να δημιουργήσετε πίνακες σαν αυτούς με τα στοιχεία τους μπορείτε να χρησιμοποιήσετε τις εντολές:

/\* Κατασκευάστε τους πίνακες Emp, Dept \*/

CREATE TABLE Dept(DEPTNO NUMBER(2) NOT NULL,

 DNAME VARCHAR2(14), LOC VARCHAR2(14),

 PRIMARY KEY(DEPTNO));

CREATE TABLE Emp(EMPNO NUMBER(4) NOT NULL,

 ENAME VARCHAR2(10), JOB VARCHAR2(9), MGR NUMBER(4),

 HIREDATE DATE, SAL NUMBER(7,2), COMM NUMBER(7,2),

 DEPTNO NUMBER(2), PRIMARY KEY(EMPNO),

 FOREIGN KEY(DEPTNO) REFERENCES Dept(DEPTNO));

/\* Εισαγωγή στοιχείων στους πίνακες \*/

INSERT INTO Dept(DEPTNO, DNAME, LOC)

 VALUES (10, 'ACCOUNTING', 'NEW YORK');

INSERT INTO Dept(DEPTNO, DNAME, LOC)

 VALUES (20, 'RESEARCH', 'DALLAS');

INSERT INTO Dept(DEPTNO, DNAME, LOC)

 VALUES (30, 'SALES', 'CHICAGO');

INSERT INTO Dept(DEPTNO, DNAME, LOC)

 VALUES (40, 'OPERATIONS', 'BOSTON');

INSERT INTO Emp(EMPNO, ENAME, JOB, MGR, HIREDATE, SAL, COMM, DEPTNO)

 VALUES (7369, 'SMITH', 'CLERK', 7902, '17/12/1980', 800, NULL, 20);

INSERT INTO Emp(EMPNO, ENAME, JOB, MGR, HIREDATE, SAL, COMM, DEPTNO)

 VALUES (7499, 'ALLEN', 'SALESMAN', 7698, '20/02/1981', 1600, 300, 30);

INSERT INTO Emp(EMPNO, ENAME, JOB, MGR, HIREDATE, SAL, COMM, DEPTNO)

 VALUES (7521, 'WARD', 'SALESMAN', 7698, '22/02/1981', 1250, 500, 30);

INSERT INTO Emp(EMPNO, ENAME, JOB, MGR, HIREDATE, SAL, COMM, DEPTNO)

 VALUES (7566, 'JONES', 'MANAGER', 7839, '02/04/1981', 2975, NULL, 20);

INSERT INTO Emp(EMPNO, ENAME, JOB, MGR, HIREDATE, SAL, COMM, DEPTNO)

 VALUES (7654, 'MARTIN', 'SALESMAN', 7698, '28/10/1981', 1250, 1400, 30);

INSERT INTO Emp(EMPNO, ENAME, JOB, MGR, HIREDATE, SAL, COMM, DEPTNO)

 VALUES (7698, 'BLAKE', 'MANAGER', 7839, '01/05/1981', 2850, NULL, 30);

INSERT INTO Emp(EMPNO, ENAME, JOB, MGR, HIREDATE, SAL, COMM, DEPTNO)

 VALUES (7782, 'CLARK', 'MANAGER', 7839, '09/06/1981', 2450, NULL, 10);

INSERT INTO Emp(EMPNO, ENAME, JOB, MGR, HIREDATE, SAL, COMM, DEPTNO)

 VALUES (7788, 'SCOTT', 'ANALYST', 7566, '19/04/1987', 3000, NULL, 20);

INSERT INTO Emp(EMPNO, ENAME, JOB, MGR, HIREDATE, SAL, COMM, DEPTNO)

 VALUES (7839, 'KING', 'PRESIDENT', NULL, '17/11/1981', 5000, NULL, 10);

INSERT INTO Emp(EMPNO, ENAME, JOB, MGR, HIREDATE, SAL, COMM, DEPTNO)

 VALUES (7844, 'TURNER', 'SALESMAN', 7698, '08/10/1981', 1500, 0, 30);

INSERT INTO Emp(EMPNO, ENAME, JOB, MGR, HIREDATE, SAL, COMM, DEPTNO)

 VALUES (7876, 'ADAMS', 'CLERK', 7788, '23/05/1987', 1100, NULL, 20);

INSERT INTO Emp(EMPNO, ENAME, JOB, MGR, HIREDATE, SAL, COMM, DEPTNO)

 VALUES (7900, 'JAMES', 'CLERK', 7698, '03/12/1981', 950, NULL, 30);

INSERT INTO Emp(EMPNO, ENAME, JOB, MGR, HIREDATE, SAL, COMM, DEPTNO)

 VALUES (7902, 'FORD', 'ANALYST', 7566, '03/12/1981', 3000, NULL, 20);

INSERT INTO Emp(EMPNO, ENAME, JOB, MGR, HIREDATE, SAL, COMM, DEPTNO)

 VALUES (7934, 'MILLER', 'CLERK', 7782, '23/01/1982', 1300, NULL, 10);

INSERT INTO Emp(EMPNO, ENAME, JOB, MGR, HIREDATE, SAL, COMM, DEPTNO)

 VALUES (7999, 'BATES', 'ANALYST', 7566, '23/01/2004', 1300, NULL, NULL);

/\* τέλος εντολών \*/

Επομένως, βασιζόμενοι στα παραπάνω κατασκευάστε αρχικά τους πίνακες MY\_EMP, MY\_DEPT που έχουν ανάλογη γραμμογράφηση και στοιχεία με τους EMP, DEPT.

CREATE TABLE my\_emp(empno NUMBER(4) NOT NULL,

 ename VARCHAR2(10),

 deptno NUMBER(2));

Αν θέλετε με την επόμενη εντολή μπορείτε να «φορτώσετε» από τον πίνακα emp δεδομένα στον νέο πίνακα.

INSERT INTO my\_emp

SELECT empno,ename,deptno FROM emp WHERE empno > 1000;

CREATE TABLE my\_dept(deptno NUMBER(2) NOT NULL,

 dname VARCHAR2(14));

Ο επόμενος trigger, είναι αποθηκευμένος (stored) και εκτελείται πριν από την εισαγωγή / ενημέρωση των στοιχείων του πίνακα my\_dept και μεταγράφει τα στοιχεία αυτά με κεφαλαία γράμματα.

CREATE OR REPLACE TRIGGER dept\_insert\_update

BEFORE INSERT OR UPDATE ON my\_dept

FOR EACH ROW

BEGIN

 DBMS\_OUTPUT.PUT\_LINE('Εκτέλεση σκανδαλισμού dept\_insert\_update');

 :NEW.dname := UPPER(:NEW.dname);

 /\* :NEW.dname contains the new value \*/

END;

/

Ακολουθούν δοκιμές.

INSERT INTO my\_dept VALUES(10, 'Learn');

INSERT INTO my\_dept VALUES(20, 'Discover');

INSERT INTO my\_dept VALUES(30, 'Explore');

SELECT \* FROM my\_dept;

UPDATE my\_dept

SET dname = 'Research'

WHERE deptno = 30;

SELECT \* FROM my\_dept;

/

Στη συνέχεια βλέπουμε τα στοιχεία του πίνακα my\_emp και προσθέτουμε στον πίνακα my\_dept στήλη που έχει τον αριθμό των υπαλλήλων για κάθε τμήμα.

SELECT \* FROM my\_emp;

ALTER TABLE my\_dept

 ADD (no\_of\_employees NUMBER(4));

Αρχικοποιούμε τη νέα στήλη.

UPDATE my\_dept

SET no\_of\_employees = (SELECT COUNT(\*)

 FROM my\_emp

 WHERE my\_emp.deptno = my\_dept.deptno);

και βλέπουμε τα στοιχεία του πίνακα.

SELECT \* FROM my\_dept;

**Στη συνέχεια γράφουμε τρεις triggers για να αυτοματοποιήσουμε τη διαχείριση της νέας στήλης του πίνακα my\_dept όταν έχουμε μεταβολή στον πίνακα my\_emp.**

CREATE OR REPLACE TRIGGER emp\_insert

AFTER INSERT ON my\_emp

FOR EACH ROW

BEGIN

 UPDATE my\_dept

 SET no\_of\_employees = no\_of\_employees + 1

 WHERE deptno = :NEW.deptno;

 /\* :NEW.deptno contains the new value

 :OLD.deptno contains NULL value \*/

END;

/

CREATE OR REPLACE TRIGGER emp\_delete

AFTER DELETE ON my\_emp

FOR EACH ROW

BEGIN

 UPDATE my\_dept

 SET no\_of\_employees = no\_of\_employees - 1

 WHERE deptno = :OLD.deptno;

 /\* :OLD.deptno contains the old value

 :NEW.deptno contains NULL value \*/

END;

/

CREATE OR REPLACE TRIGGER emp\_update

AFTER UPDATE ON my\_emp

FOR EACH ROW

BEGIN

/\* :NEW.deptno contains the new value

 :OLD.deptno contains the old value \*/

/\* Increments the count for the employee's new department \*/

 UPDATE my\_dept

 SET no\_of\_employees = NVL(no\_of\_employees,0) + 1

 WHERE deptno = :NEW.deptno;

/\* Decreases the count for the employee's old department \*/

 UPDATE my\_dept

 SET no\_of\_employees = no\_of\_employees - 1

 WHERE deptno = :OLD.deptno;

END;

/

Ακολουθούν δοκιμές / έλεγχοι.

INSERT INTO my\_dept VALUES(11,'Belle Epoque',0);

INSERT INTO my\_dept VALUES(12,'Rasors edge',0);

INSERT INTO my\_emp(empno,ename,deptno) VALUES (102,'Luers',11);

INSERT INTO my\_emp(empno,ename,deptno) VALUES (103,'Atwood',11);

INSERT INTO my\_emp(empno,ename,deptno) VALUES (104,'Gennick',12);

SELECT \* FROM my\_dept WHERE deptno IN (11,12);

DELETE FROM my\_emp WHERE empno = 103;

SELECT \* FROM my\_dept WHERE deptno IN (11,12);

UPDATE my\_emp

SET deptno = 11

WHERE empno = 104;

SELECT \* FROM my\_dept WHERE deptno IN (11,12);

/

## Κάποιες άλλες χρήσιμες υποδείξεις για τη διαχείριση των triggers στο προϊόν της Oracle

Πως θα δούμε τη γραμμογράφηση και κάποια στοιχεία πινάκων του συστήματος που σχετίζονται με τους triggers.

DESCRIBE USER\_TRIGGERS;

SELECT trigger\_name,triggering\_event,trigger\_type

FROM USER\_TRIGGERS

WHERE table\_name = 'MY\_EMP'

ORDER BY trigger\_name;

/

SELECT trigger\_name,trigger\_body

FROM USER\_TRIGGERS

WHERE table\_name= 'MY\_EMP'

ORDER BY trigger\_name;

/

Πως θα περιορίσουμε τη χρήση της βάσης για συγκεκριμένες ώρες.

CREATE OR REPLACE TRIGGER only\_during\_my\_hours

BEFORE INSERT OR UPDATE OR DELETE ON my\_emp

BEGIN

 DBMS\_OUTPUT.PUT\_LINE('only\_during\_my\_hours trigger');

 IF TO\_NUMBER(TO\_CHAR(SYSDATE,'hh24')) < 8

/\* nothing before 8:00 am \*/

 OR TO\_NUMBER(TO\_CHAR(SYSDATE,'hh24')) >= 5

/\* changes must be made before 5:00 pm \*/

 OR TO\_CHAR(SYSDATE,'dy') IN ('sun') THEN

/\* nothing on Sunday \*/

 RAISE\_APPLICATION\_ERROR(-20000,

 'Αλλαγές μόνο τις ώρες που θέλουμε');

 END IF;

END;

/

|  |
| --- |
| **Ανοικτά Ακαδημαϊκά Μαθήματα****Τεχνολογικό Εκπαιδευτικό Ίδρυμα Αθήνας** |
| **Τέλος Ενότητας** |
| **Χρηματοδότηση*** Το παρόν εκπαιδευτικό υλικό έχει αναπτυχθεί στα πλαίσια του εκπαιδευτικού έργου του διδάσκοντα.
* Το έργο «**Ανοικτά Ακαδημαϊκά Μαθήματα στο ΤΕΙ Αθήνας**» έχει χρηματοδοτήσει μόνο τη αναδιαμόρφωση του εκπαιδευτικού υλικού.
* Το έργο υλοποιείται στο πλαίσιο του Επιχειρησιακού Προγράμματος «Εκπαίδευση και Δια Βίου Μάθηση» και συγχρηματοδοτείται από την Ευρωπαϊκή Ένωση (Ευρωπαϊκό Κοινωνικό Ταμείο) και από εθνικούς πόρους.

 |

**Σημειώματα**

**Σημείωμα Αναφοράς**

Copyright ΤΕΙ Αθήνας, Χ. Σκουρλάς, Α. Τσολακίδης, Δ. Βάσσης 2014. Χ. Σκουρλάς, Α. Τσολακίδης, Δ. Βάσσης. «Βάσεις Δεδομένων ΙΙ (Ε). Ενότητα 14: Επισκόπηση της Διαχείρισης triggers στο περιβάλλον mySQL και σε Oracle». Έκδοση: 1.0. Αθήνα 2014. Διαθέσιμο από τη δικτυακή διεύθυνση: [ocp.teiath.gr](https://ocp.teiath.gr/).

**Σημείωμα Αδειοδότησης**

Το παρόν υλικό διατίθεται με τους όρους της άδειας χρήσης Creative Commons Αναφορά, Μη Εμπορική Χρήση Παρόμοια Διανομή 4.0 [1] ή μεταγενέστερη, Διεθνής Έκδοση. Εξαιρούνται τα αυτοτελή έργα τρίτων π.χ. φωτογραφίες, διαγράμματα κ.λ.π., τα οποία εμπεριέχονται σε αυτό. Οι όροι χρήσης των έργων τρίτων επεξηγούνται στη διαφάνεια «Επεξήγηση όρων χρήσης έργων τρίτων».

Τα έργα για τα οποία έχει ζητηθεί άδεια αναφέρονται στο «Σημείωμα Χρήσης Έργων Τρίτων».



[1] http://creativecommons.org/licenses/by-nc-sa/4.0/

Ως **Μη Εμπορική** ορίζεται η χρήση:

* που δεν περιλαμβάνει άμεσο ή έμμεσο οικονομικό όφελος από την χρήση του έργου, για το διανομέα του έργου και αδειοδόχο
* που δεν περιλαμβάνει οικονομική συναλλαγή ως προϋπόθεση για τη χρήση ή πρόσβαση στο έργο
* που δεν προσπορίζει στο διανομέα του έργου και αδειοδόχο έμμεσο οικονομικό όφελος (π.χ. διαφημίσεις) από την προβολή του έργου σε διαδικτυακό τόπο

Ο δικαιούχος μπορεί να παρέχει στον αδειοδόχο ξεχωριστή άδεια να χρησιμοποιεί το έργο για εμπορική χρήση, εφόσον αυτό του ζητηθεί.

**Επεξήγηση όρων χρήσης έργων τρίτων**

|  |  |
| --- | --- |
| © | Δεν επιτρέπεται η επαναχρησιμοποίηση του έργου, παρά μόνο εάν ζητηθεί εκ νέου άδεια από το δημιουργό. |
| διαθέσιμο με άδεια CC-BY | Επιτρέπεται η επαναχρησιμοποίηση του έργου και η δημιουργία παραγώγων αυτού με απλή αναφορά του δημιουργού. |
| διαθέσιμο με άδεια CC-BY-SA | Επιτρέπεται η επαναχρησιμοποίηση του έργου με αναφορά του δημιουργού, και διάθεση του έργου ή του παράγωγου αυτού με την ίδια άδεια. |
| διαθέσιμο με άδεια CC-BY-ND | Επιτρέπεται η επαναχρησιμοποίηση του έργου με αναφορά του δημιουργού. Δεν επιτρέπεται η δημιουργία παραγώγων του έργου. |
| διαθέσιμο με άδεια CC-BY-NC | Επιτρέπεται η επαναχρησιμοποίηση του έργου με αναφορά του δημιουργού. Δεν επιτρέπεται η εμπορική χρήση του έργου. |
| διαθέσιμο με άδεια CC-BY-NC-SA | Επιτρέπεται η επαναχρησιμοποίηση του έργου με αναφορά του δημιουργού και διάθεση του έργου ή του παράγωγου αυτού με την ίδια άδεια. Δεν επιτρέπεται η εμπορική χρήση του έργου. |
| διαθέσιμο με άδεια CC-BY-NC-ND | Επιτρέπεται η επαναχρησιμοποίηση του έργου με αναφορά του δημιουργού. Δεν επιτρέπεται η εμπορική χρήση του έργου και η δημιουργία παραγώγων του. |
| διαθέσιμο με άδεια CC0 Public Domain | Επιτρέπεται η επαναχρησιμοποίηση του έργου, η δημιουργία παραγώγων αυτού και η εμπορική του χρήση, χωρίς αναφορά του δημιουργού. |
| διαθέσιμο ως κοινό κτήμα | Επιτρέπεται η επαναχρησιμοποίηση του έργου, η δημιουργία παραγώγων αυτού και η εμπορική του χρήση, χωρίς αναφορά του δημιουργού. |
| χωρίς σήμανση | Συνήθως δεν επιτρέπεται η επαναχρησιμοποίηση του έργου. |

**Διατήρηση Σημειωμάτων**

* Οποιαδήποτε αναπαραγωγή ή διασκευή του υλικού θα πρέπει να συμπεριλαμβάνει:
* το Σημείωμα Αναφοράς
* το Σημείωμα Αδειοδότησης
* τη δήλωση Διατήρησης Σημειωμάτων
* το Σημείωμα Χρήσης Έργων Τρίτων (εφόσον υπάρχει) μαζί με τους συνοδευόμενους υπερσυνδέσμους.